

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

THIS PAGE BLANK (USPTO)

PCT

WELTORGANISATION FÜR GEISTIGES EIGENTUM
Internationales Büro



INTERNATIONALE ANMELDUNG VERÖFFENTLICHT NACH DEM VERTRAG ÜBER DIE
INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS (PCT)

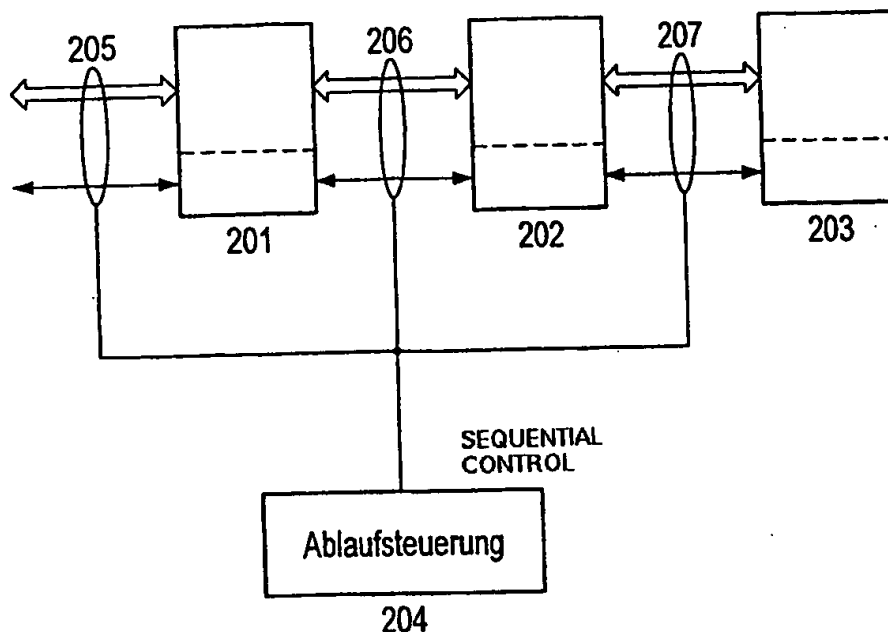
(51) Internationale Patentklassifikation ⁶ : G06F 17/50		A1	(11) Internationale Veröffentlichungsnummer: WO 99/60497
(21) Internationales Aktenzeichen: PCT/DE99/01324		(43) Internationales Veröffentlichungsdatum: 25. November 1999 (25.11.99)	
(22) Internationales Anmeldedatum: 3. Mai 1999 (03.05.99)		(81) Bestimmungsstaaten: US, europäisches Patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).	
(30) Prioritätsdaten: 198 22 502.4 19. Mai 1998 (19.05.98) DE		Veröffentlicht <i>Mit internationalem Recherchenbericht.</i> <i>Vor Ablauf der für Änderungen der Ansprüche zugelassenen Frist; Veröffentlichung wird wiederholt falls Änderungen eintreffen.</i>	
(71) Anmelder (für alle Bestimmungsstaaten ausser US): SIEMENS AKTIENGESELLSCHAFT [DE/DE]; Wittelsbacherplatz 2, D-80333 München (DE).			
(72) Erfinder; und (75) Erfinder/Anmelder (nur für US): ROSEN, Roland [DE/DE]; Am Ländtbogen 6, D-82211 Herrsching (DE). WÖLLHAF, Konrad [DE/DE]; Am Europakanal 8, D-91056 Erlangen (DE).			
(74) Gemeinsamer Vertreter: SIEMENS AKTIENGESELLSCHAFT; Postfach 22 16 34, D-80506 München (DE).			

(54) Title: METHOD FOR CARRYING OUT THE COMPUTER-AIDED SIMULATION OF A TECHNICAL SYSTEM

(54) Bezeichnung: VERFAHREN ZUR RECHNERGESTÜTZTEN SIMULATION EINES TECHNISCHEN SYSTEMS

(57) Abstract

The invention relates to a method for simulating a real technical system with the aid of a computer. Models are produced of individual components of the technical system as components of the simulation. Each component has an interface for a flow of process material and an information technology interface. The components are controlled by a main process which also ensures their interaction with each other to reproduce the system being simulated as a whole. Functionalities of the real system are encapsulated in the components, whilst the flow of material and the information technology interface determine the behaviour of each component towards the outside world.



(57) Zusammenfassung

Eine reale technische Anlage wird rechnergestützt simuliert, wobei einzelne Anlagenkomponenten der technischen Anlage als Komponenten der Simulation modelliert werden und jede Komponente eine Schnittstelle für einen verfahrenstechnischen Stofffluß und eine informationstechnische Schnittstelle aufweist. Die Steuerung und die Interaktion der Komponenten untereinander zur Gewährleistung des zu simulierenden Gesamtsystems wird durch einen Hauptprozeß gewährleistet. Funktionalitäten des realen technischen Systems sind dabei in den Komponenten gekapselt, wobei der Stofffluß und die informationstechnische Schnittstelle das Verhalten jeder Komponente mit der Außenwelt bestimmen.

LEDIGLICH ZUR INFORMATION

Codes zur Identifizierung von PCT-Vertragsstaaten auf den Kopfbögen der Schriften, die internationale Anmeldungen gemäss dem PCT veröffentlichen.

AL	Albanien	ES	Spanien	LS	Lesotho	SI	Slowenien
AM	Armenien	FI	Finnland	LT	Litauen	SK	Slowakei
AT	Österreich	FR	Frankreich	LU	Luxemburg	SN	Senegal
AU	Australien	GA	Gabun	LV	Lettland	SZ	Swasiland
AZ	Aserbaidshan	GB	Vereinigtes Königreich	MC	Monaco	TD	Tschad
BA	Bosnien-Herzegowina	GE	Georgien	MD	Republik Moldau	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagaskar	TJ	Tadschikistan
BE	Belgien	GN	Guinea	MK	Die ehemalige jugoslawische Republik Mazedonien	TM	Turkmenistan
BF	Burkina Faso	GR	Griechenland	ML	Mali	TR	Türkei
BG	Bulgarien	HU	Ungarn	MN	Mongolei	TT	Trinidad und Tobago
BJ	Benin	IE	Irland	MR	Mauretanien	UA	Ukraine
BR	Brasilien	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Island	MX	Mexiko	US	Vereinigte Staaten von Amerika
CA	Kanada	IT	Italien	NE	Niger	UZ	Usbekistan
CF	Zentralafrikanische Republik	JP	Japan	NL	Niederlande	VN	Vietnam
CG	Kongo	KE	Kenia	NO	Norwegen	YU	Jugoslawien
CH	Schweiz	KG	Kirgisistan	NZ	Neuseeland	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Demokratische Volksrepublik Korea	PL	Polen		
CM	Kamerun	KR	Republik Korea	PT	Portugal		
CN	China	KZ	Kasachstan	RO	Rumänien		
CU	Kuba	LC	St. Lucia	RU	Russische Föderation		
CZ	Tschechische Republik	LI	Liechtenstein	SD	Sudan		
DE	Deutschland	LK	Sri Lanka	SE	Schweden		
DK	Dänemark	LR	Liberia	SG	Singapur		
EE	Estland						

Beschreibung**Verfahren zur rechnergestützten Simulation eines technischen Systems**

5

Die Erfindung betrifft ein Verfahren zur rechnergestützten Simulation eines technischen Systems, welches System mehrere Komponenten umfaßt.

10 In der Verfahrenstechnik ist eine große Herausforderung die Planung komplexer technischer Systeme. Derartige Systeme sind insbesondere große Anlagen für die Fertigung oder die Koordination chemischer und/oder physikalischer Prozesse.

15 Die Umsetzung einer fehlerhaften Planung führt unter Umständen zu hohen Zusatzausgaben, ehe ein reibungsloser Ablauf der Anlage gewährleistet werden kann. Ein Planungsfehler stellt eine signifikante Größe in den Kosten für die Anlage dar.

20

Begriffe der objektorientierten Programmierung sind aus [1] bekannt. Insbesondere wird eine Klasse instantiiert und dabei ein Objekt vom Typ der Klasse geschaffen. Auf diese Art können beliebig viele Objekte des gleichen Typs, also mit
25 jeweils gleicher Funktionalität, instantiiert werden. Dabei ist die Funktionalität in dem Objekt verborgen (gekapselt), der Zugriff von außen bzw. die Mitteilung nach außen bleibt findet über vorgegebene Schnittstellen, insbesondere durch Methodenaufrufe, statt. Weiterhin ist eine hierarchische

30 Struktur von Typen möglich, die auf eine gemeinsame Funktionalität über den Mechanismus der Vererbung zurückgreifen.

Die **Aufgabe** der Erfindung besteht darin, ein Verfahren zur
35 rechnergestützten Simulation eines technischen Systems anzugeben, welche Simulation vor der Umsetzung der Planung

eine deutliche Reduzierung der Planungsfehler und der damit verbundenen Kosten sicherstellt.

5 Diese Aufgabe wird gemäß den Merkmalen des unabhängigen Patentanspruchs gelöst.

Die rechnergestützte Simulation findet insbesondere in Anlehnung an reale Anlagenkomponenten statt. Nachfolgend wird insbesondere von einer Komponente als eine Abbildung der
10 realen Anlagenkomponente ausgegangen.

Es wird ein Verfahren zur rechnergestützten Simulation eines technischen Systems angegeben, welches (simulierte) System mehrere Komponenten umfaßt. Dabei enthält jede Komponente
15 mindestens eine Schnittstelle, und jeder Komponente ist eine Funktionalität einbeschrieben. Die Simulation des technischen Systems wird durchgeführt, indem die mehreren Komponenten zusammengefügt werden und über die mindestens eine Schnittstelle Information zwischen den Komponenten
20 ausgetauscht wird, wobei anhand eines Hauptprozesses ein Zusammenwirken der mehreren Komponenten gesteuert und ausgewertet wird.

Der Hauptprozeß stellt insbesondere ein Interagieren der
25 einzelnen Komponenten und somit einen Simulationsablauf sicher, indem eine Verwaltung einer simulierten Zeit (die Zeit, die in der realen Nachbildung des simulierten technischen Systems vergehen würde) von dem Hauptprozeß gewährleistet wird.

30 Eine Weiterbildung der Erfindung besteht darin, daß eine Komponente ein Anlagenobjekt ist. Das Anlagenobjekt ist ein Teil des zu simulierenden technischen Systems.

35 Auch ist es eine Weiterbildung, daß die Komponente eine Instanz einer Klasse eines objektorientierten Programms ist,

durch welche Klasse der Typ des Anlagenobjekts beschrieben wird.

Gemäß den Möglichkeiten der objektorientierten Programmierung (vgl. [1]) ermöglicht die Instantiierung einer Klasse die Erschaffung eines (simulierten) Anlagenobjekts, wobei alle einbeschriebene Funktionalität des Anlagenobjekts in dieser Instanz verfügbar ist. Benötigt man mehrere Anlagenobjekte dieses Typs, so können diese mehrfach instantiiert werden.

Hierarchische Typen von Anlagenobjekten, die unterschiedliche Funktionalitäten aufweisen, werden über den Mechanismus der Vererbung erzeugt. Die Funktionalität eines Anlagenobjekts ist in dem jeweiligen Anlagenobjekt gekapselt und von außen nur über vorgegebene Schnittstellen erreichbar. Auf diese Weise können viele Anlagenobjekte als ein neues Anlagenobjekt zusammengefaßt und als Teil des technischen Systems bereitgestellt werden. Gemäß der Erklärung aus [1] wird der Typ des Anlagenobjekts als eine Klasse vereinbart, eine Instanz stellt einen tatsächlichen Repräsentanten der Klasse dar.

Ein Vorteil ist darin zu sehen, daß lediglich im Zusammenschluß mehrerer Komponenten, die vorzugsweise Anlagenobjekte darstellen, eine Simulation eines vollständigen technischen Systems in Koordination durch einen Hauptprozeß ermöglicht wird.

Hierbei sei angemerkt, daß neben der objektorientierten Programmierung auch eine prozedurale Programmierung des Verfahrens möglich ist. Allerdings stellt die objektorientierte Programmierung bereits Funktionalitäten bereit, deren Verwendung die beschriebenen Vorteile begründet.

Nachfolgend werden einige Begriffe, die im Zusammenhang mit der vorliegenden Erfindung genannt sind, erläutert.

- Technisches System:

Unter einem technischen System wird eine mehr oder minder komplexe technische Anlage, vorzugsweise aus dem Bereich der Verfahrenstechnik, verstanden, wobei jede Anlage wiederum ein Teil einer übergeordneten - dementsprechend größeren - Anlage sein kann. Dieser hierarchischen Strukturierung wird insbesondere durch den Mechanismus der Kapselung der Funktionalität des Anlagenobjekts Rechnung getragen.

- Komponente:

Die Komponente ist insbesondere ein Anlagenobjekt, z.B. Motor, Pumpe, Ventil, Rohrleitung, Behälter, Formstück, Regler (PID, PI, P), Werkzeugmaschine, Lager, Stromrichter, Trafo, Generator, Getriebe, Propeller, Sammelschiene, Leistungsschalter, Hydraulik, und wird bevorzugt durch eine geeignete Beschreibungsform modelliert. Bei der Beschreibung der Komponente wird insbesondere geachtet auf die zu simulierende Vorgabe, d.h. die Funktionalität der Komponente wird anhand der Beschreibungsform, vorzugsweise durch Gleichungssysteme, nachgebildet.

Die oben genannten Beispiele für Komponenten stellen keine vollständige Menge aller möglichen Komponenten dar, sondern deuten vielmehr an, inwieweit unterschiedliche Komponenten modelliert werden können.

- Hauptprozeß:

Der Hauptprozeß (auch: Ablaufsteuerung) gewährleistet durch geeignete Ansteuerung der Komponenten deren funktionales Zusammenwirken. Ferner wertet der Hauptprozeß die Ergebnisse der einzelnen Komponenten aus und stellt sie einem Benutzer vorzugsweise graphisch dar.

- Simulationsrechenzeit:

Eine Zeit, die zur Durchführung der Simulation benötigt wird.

Eine andere Weiterbildung besteht darin, daß die mindestens eine Schnittstelle eine erste Teilschnittstelle und eine zweite Teilschnittstelle umfaßt. Dabei weist die erste Teilschnittstelle einen Eingang und einen Ausgang auf, wobei
5 über den Eingang und den Ausgang der ersten Teilschnittstelle ein Stoff-Fluß modelliert wird. Die zweite Teilschnittstelle weist ebenfalls einen Eingang und einen Ausgang auf, wobei durch den Eingang der zweiten Teilschnittstelle eine
10 Veränderung einer Stellgröße der Komponente durchgeführt wird und durch den Ausgang der zweiten Teilschnittstelle eine Rückmeldung über den Zustand des Stoff-Flusses und/oder über einen Zustand der Veränderung der Stellgröße ermittelbar ist.

Somit umfaßt die Komponente, insbesondere das Anlagenobjekt,
15 einen in der Verfahrenstechnik relevanten Stoff-Fluß (erste Teilschnittstelle) und eine informationstechnische Schnittstelle (zweite Teilschnittstelle). Die informationstechnische Schnittstelle wird insbesondere verwendet zur Steuerung einer Stellgröße der jeweiligen
20 Komponente und Ermittlung eines Soll/Ist-Unterschieds zwischen der eingestellten Stellgröße und der sich ergebenden Stellgröße.

Die Komponente, insbesondere das Anlagenobjekt, wird durch
25 mindestens eine der folgenden Beschreibungsformen dargestellt:

a) Differentialalgebraisches System:

Ein differentialalgebraisches System umfaßt sowohl
30 algebraische Gleichungssysteme als auch Differentialgleichungssysteme, um die vorgegebene Funktionalität des Anlagenobjekts zu beschreiben. Ein Beispiel für ein differentialalgebraisches System ist die Modellierung eines Behälters:

35

$$\frac{\partial F}{\partial t} = q_{zu} - q_{ab};$$

$$p = \frac{F}{A} \cdot \rho \cdot g,$$

wobei

- 5 F die Füllmenge,
 t die Zeit,
 q_{zu} den Zufluß in den Behälter,
 q_{ab} den Abfluß aus dem Behälter,
 p den Bodendruck in dem Behälter,
10 A die Grundfläche des Behälters,
 ρ die Dichte der Flüssigkeit und
 g die Erdbeschleunigung
 bezeichnen.

15 b) Ereignisdiskretes Modell:

Unter einem ereignisdiskreten Modell versteht man insbesondere eine zeitgetriggerte Modellierung. Basierend auf Zeiteinheiten erfolgt eine Nachrichtenübermittlung an den Hauptprozeß oder an eine andere Komponente, um dort
20 eine mit einem Auslöse-Zeitpunkt verknüpfte Bearbeitung anzustoßen.

c) Strukturinformation:

Schnittstellen der Komponenten weisen insbesondere eine
25 Strukturinformation auf. Anhand dieser Strukturinformation ist es möglich, Eigenschaften, die sich aus der Struktur des technischen Systems (bzw. des Gesamtsystems) ergeben, geeignet zu modellieren.

Einem Zustand einer Komponente entsprechend werden die
30 Eigenschaften der Schnittstellen gesetzt, diese Eigenschaften an Schnittstellen anderer Komponenten durchgereicht, abgefragt oder blockiert (nicht weitergegeben). Die Strukturinformation läßt sich im Rahmen der Modellierung auf verschiedene Art und Weise
35 verwenden.

Beispielsweise gibt ein geschlossenes Ventil die

Struktureigenschaft "Druck ist gesetzt" nicht an eine benachbarte Komponente weiter. Ein Behälter hingegeben legt den Druck an seiner Schnittstelle fest. Befindet sich eine Komponente zwischen zwei geschlossenen Ventilen, ist diese spezielle Situation durch die Strukturinformation innerhalb der Komponente bekannt und kann bei der Modellierung berücksichtigt werden. Dementsprechend stellt die Komponente einen Ersatzwert für den Druck bereit, der in die Modellierung einfließt.

Die der Komponente innewohnende Strukturinformation wird von dem Hauptprozeß derart berücksichtigt, daß diese Strukturinformation über Komponenten hinweg entsprechend der vorgegebenen Verbindung der Komponenten kommuniziert wird und somit potentielle Konflikte der Simulation aufgelöst werden.

Eine andere Weiterbildung besteht darin, daß durch die erste Schnittstelle ein Prozeßverhalten und durch die zweite Schnittstelle ein Kontrollverhalten modelliert wird.

Unter dem Prozeßverhalten versteht man die Abbildung des in der Verfahrenstechnik üblichen Stoff-Flusses auf das Modell, wobei ein Kontrollverhalten der informationstechnischen Auslegung von Soll/Ist-Wert der Stellgrößen einer Komponente gleichkommt.

Auch ist es eine Weiterbildung, daß die Komponente über mindestens einen der folgenden Mechanismen Nachrichten erzeugt:

- a) Von der Komponente selbst wird ein Ereignis (engl. Fachbegriff: Event) generiert und an den Hauptprozeß übermittelt;
- b) von der Komponente wird eine Variablenänderung mit der Adresse einer Zielkomponente generiert und an den Hauptprozeß übermittelt;

- c) von der Komponente wird eine Methode einer anderen Komponente aufgerufen, indem die Bezeichnung der Methode mit der Adresse der Zielkomponente generiert und an den Hauptprozeß übermittelt wird.

Im Rahmen einer Ausgestaltung umfaßt der Hauptprozeß folgende Schritte:

- a) Das Ereignis wird in einer Warteschlange abgespeichert.
- b) Wenn das Ereignis fällig ist, d.h. wenn der Zeitpunkt der Fälligkeit in der simulierten Zeit erreicht ist, wird dieses Ereignis an den Adressaten weitergeleitet und dort ausgeführt; das Ereignis wird in diesem Fall aus der Warteschlange gelöscht.

Die Verwaltung der Warteschlange liegt beim Hauptprozeß.

- Eine andere Weiterbildung besteht darin, daß mindestens eine Schnittstelle jeder Komponente derart ausgeführt ist, daß mehrere Komponenten entsprechend ihrer zugrundeliegenden technischen Bedeutung einfach zusammengefügt werden können. Dabei ist insbesondere die Modularität der einzelnen Komponenten von Vorteil, die über vorgegebene Schnittstellen miteinander in Verbindung treten und einen Simulationsablauf entsprechend einem technischen System, das sich aus der Art und Weise der Verbindung der Komponenten untereinander ergibt, gewährleistet ist.

- Insbesondere sei an dieser Stelle darauf hingewiesen, daß der in vorliegendem Dokument beschriebene Stoff-Fluß unabhängig von der im Modell angegebenen Richtung modelliert werden kann. Somit bedingen Eingang und Ausgang einer Komponente nicht die Richtung des Stoff-Flusses. Bei Eingang und Ausgang handelt es sich lediglich um bei der Modellierung bezeichnete Schnittstellen der jeweiligen Komponente.

Weiterbildungen der Erfindung ergeben sich auch aus den abhängigen Ansprüchen.

- 5 Ausführungsbeispiele der Erfindung werden nachfolgend anhand der Zeichnung dargestellt und erläutert.

Es zeigen

- 10 Fig.1 eine Komponente, die einen Teil des zu simulierenden Systems darstellt;
- Fig.2 einen Verbund aus mehreren Komponenten, die von einer Ablaufsteuerung verwaltet werden;
- 15 Fig.3 ein Flußdiagramm, das Schritte einer Ablaufsteuerung darstellt;
- Fig.4 ein Nachrichtenformat, das zur Kommunikation zwischen Komponenten bzw. zwischen Komponente und Hauptprozeß verwendet wird;
- 20 Fig.5 eine Warteschlange, die von der Ablaufsteuerung bearbeitet wird;
- 25 Fig.6 eine verfahrenstechnische Anlage für ein Drei-Tank-Beispiel.
- 30 Fig.1 zeigt eine Skizze einer Komponente, der eine zu simulierende Funktionalität einbeschrieben ist. Die Komponente 101 umfaßt eine erste Teilschnittstelle (102, 103), die einen Eingang 102 und einen Ausgang 103 aufweist, und einen Stoff-Fluß modelliert. Eine zweite
- 35 Teilschnittstelle (104, 105) umfaßt einen Eingang 104, welcher Eingang 104 eine Veränderung einer Stellgröße der Komponente 101 ermöglicht, und einen Ausgang 105, welcher

Ausgang 105 eine Rückmeldung über einen Zustand des Stoff-Flusses und/oder über einen Zustand der Veränderung der Stellgröße ermöglicht.

- 5 Eine derartige Komponente 101 stellt eine modulare Einheit eines zu simulierenden technischen Systems dar. Über die Schnittstellen (102-105) wird die Komponente 101 mit weiteren Komponenten verbunden. In dem Zusammenschluß mehrerer Komponenten besteht das technische System, das zu simulieren
- 10 ein Ziel der Erfindung darstellt. Durch die normierten Schnittstellen können Komponenten, denen unterschiedliche Funktionalitäten einbeschrieben sind, gesteuert durch den Hauptprozeß, interagieren.
- 15 Fig.2 zeigt einen möglichen Zusammenschluß dreier Komponenten (201, 202, 203) deren Schnittstellen (205, 206, 207) jeweils mit einer Ablaufsteuerung 204 verbunden sind. Die Ablaufsteuerung 204 gewährleistet die Durchführung der Simulation, wobei sowohl eine Kommunikation zwischen den
- 20 Komponenten als auch eine Kommunikation zwischen einer Komponente und der Ablaufsteuerung erfolgt. Die Ablaufsteuerung 204 steuert die Komponenten über die informationstechnische Schnittstelle (zweite Teilschnittstelle) und überträgt das Prozeßverhalten (Stoff-
- 25 Fluß über erste Teilschnittstelle) unter Berücksichtigung einer simulierten Zeit von einer Komponente zur nächsten, wobei ein Einfluß auf das Prozeßverhalten in einer jeweiligen Komponente entsprechend der der Komponente zugrundeliegenden technischen Funktionalität, berücksichtigt wird.
- 30 Zu jedem (diskreten) Zeitpunkt der simulierten Zeit erfolgt eine Abarbeitung aller zu diesem Zeitpunkt anstehenden parallelen Aktionen. Dabei werden das Prozeßverhalten (erste Teilschnittstelle) und das Kontrollverhalten (zweite Teilschnittstelle, informationstechnische Schnittstelle) für
- 35 die Beeinflussung des Stoff-Flusses der Ablaufsteuerung mitgeteilt bzw. von der Ablaufsteuerung an die betroffenen

Komponenten verteilt. Die Zuordnung von Nachrichten zu den Komponenten übernimmt die Ablaufsteuerung.

- 5 Ein Ablaufdiagramm, das die Schritte eines Hauptprozesses 301 (Ablaufsteuerung) enthält, ist in Fig.3 dargestellt. Zur Beschreibung einer Funktionalität der Komponente stehen verschiedene Beschreibungsformen (Differentialalgebraisches System, ereignisdiskretes Modell, Strukturinformation) zur
10 Auswahl, die jeweils für eine bestimmte Teilfunktionalität geeignet sind. Durch Kombination der Beschreibungsformen entsteht die Gesamtfunktionalität der Komponente. Sowohl diese Kombination als auch das Zusammenwirken verschiedener Komponenten wird vom Hauptprozeß gewährleistet. Jede
15 Beschreibungsform umfaßt ein oder mehrere Teile, die hier als (Beschreibungs-)Segmente bezeichnet werden. Durch Benennung der Segmente greift der Hauptprozeß auf die unterschiedlichen Segmente zu, kombiniert sie und bildet entsprechende Teilfunktionalitäten. In Fig.3 ist ein Segment in Form eines
20 Rechtecks dargestellt.

- Der Hauptprozeß 301 führt die Anlagensimulation für ein vorgegebenes Zeitintervall aus, das zwischen "Start intervall" 302 und "End intervall" 303 liegt. Durch
25 wiederholtes Ausführen des Hauptprozesses 301 kann eine längere Zeitdauer simuliert werden.
Zu Beginn des Zeitintervalls ruft der Hauptprozeß 301 für alle Komponenten das Segment "InputControl" 304 auf, in dem für jede Komponente die von der Simulationsumgebung
30 eingetroffenen Eingaben (z.B. Benutzereingaben, Daten anderer Programme, z.B. Planungs- oder Steuer-/Regelungsprogramme) überprüft und übernommen werden.

- Daraufhin wird vom Hauptprozeß 301 für alle Komponenten das
35 Segment "SettingProperties" 305 abgearbeitet, in der die Strukturinformation, die jede Komponente beiträgt, aufbereitet wird. Anschließend wertet der Hauptprozeß 301

diese Strukturinformation aus (vgl. Block 306) und stellt damit nachfolgenden Segmenten Ergebnisse der Auswertung zur Verfügung.

- 5 Es schließt sich nun der ereignisdiskrete Modellierungsteil 307 an, wobei dieser auf insgesamt drei Segmente aufgeteilt wird. Zunächst wird für jede Komponente einmalig ein Segment "InitDiscreteModel" 308 aufgerufen, anschließend wird wiederholt für alle Komponenten ein Segment "DiscreteModel" 10 309 ausgeführt. Der Hauptprozeß wiederholt dieses Segment "DiscreteModel" 309 für alle Komponenten so lange, wie neue Daten über die Schnittstellen, die zwischen den Komponenten bestehen, zu den Komponenten gelangen. Abgeschlossen wird dieser Modellierungsteil durch den einmaligen Aufruf des 15 Segmentes "PostDiscreteModel" 310 für alle Komponenten. Innerhalb der drei Segmente 307 können Ereignisse, die zu späteren Zeitpunkten berücksichtigt werden sollen, formuliert werden. Der Hauptprozeß speichert diese Ereignisse und stellt sie (und die mit dem Ereignis verbundenen Daten) bei 20 Erreichen des Ereigniszeitpunktes den Segmenten zur Verfügung.

In den ereignisdiskreten Segmenten können zusätzlich auch strukturelevante Daten modifiziert werden. Dies führt 25 unmittelbar zu einem Rücksprung vor die Ausführung der Komponenten-Segmente "SettingProperties" 305.

Nun wird der differentialalgebraische Modellierungsteil 311, der in insgesamt sieben Segmente unterteilt ist, 30 abgearbeitet. In den beiden Segmenten "SelectionOfVariables" 312 und "SelectionOfEquations" 313 wird hinterlegt, welche Variablen und welche Gleichungen (algebraische Gleichungen und/oder gewöhnliche Differentialgleichungen) zum aktuellen Zeitpunkt in die Modellierung eingehen sollen. Die Definition 35 der Gleichungen befindet sich in den Segmenten "G-Equations" 315 (für algebraische Gleichungen) und "F-Equations" 316 (für Differentialgleichungen). Diese Daten sammelt der Hauptprozeß

301 auf und wertet das so entstehende differentialalgebraische System aus (vgl. Block 318). Dabei wird optional das Segment "JacobiEquations" 317 verwendet, das die Erstellung des zur mathematischen Lösung des Systems erforderliche Jacobi-Matrix in minimaler Rechenzeit erlaubt. Die Auswertung bedingt, daß die simulierte Zeit voranschreitet. Der Hauptprozeß 301 kontrolliert dieses Voranschreiten und stoppt die Simulationsrechenzeit, sobald entweder ein Ereigniszeitpunkt aus dem ereignisdiskreten Modellteil oder der Endzeitpunkt des Zeitintervalls erreicht ist. Außerdem kontrolliert der Hauptprozeß 301, ob während der Auswertung eine Schaltfunktion, die im Segment "SwitchingFunctions" 314 beschrieben ist, ausgelöst wird. Mit einer Schaltfunktion kann beispielsweise geprüft werden, ob eine zeitabhängige Variable, die durch eine Differentialgleichung modelliert wird, eine bestimmte Schranke über- oder unterschritten hat. Ist dies der Fall, so wird vom Hauptprozeß 301 ein Ereignis für den aktuelle Zeitpunkt erzeugt.

Liegt (mindestens) ein Ereignis an bzw. ist der Endzeitpunkt für ein Ereignis erreicht, so wird einmalig für alle Komponenten das Segment "PostAlgebraic" 319 aufgerufen. Danach erfolgt im Falle des Anliegens eines Ereignisses ein Rücksprung vor die Auswertung des ereignisdiskreten Modellteils, in dem das (die) Ereignis(se) berücksichtigt werden.

Ist der Endzeitpunkt erreicht, so wird einmalig für alle Komponenten das Segment "PostExecute" 320 aufgerufen, in dem insbesondere Ausgaben an einen Benutzer oder an andere angeschlossene Programme erfolgen.

Nicht abgebildet in der Figur Fig. 3 sind zwei weitere Segmente, die für alle Komponenten einmalig, zum Zeitpunkt des Programmstartes der Simulation und unmittelbar vor Programmende der Simulation vom Hauptprozeß aufgerufen wird.

Diese beiden Segmente erlauben verwaltungstechnische und EDV-technische Maßnahmen (z.B. Lesen von Initialisierungsdateien, Schliessen von Dateien mit beispielsweise statistischen Zusatzausgaben).

5

Fig.4 zeigt ein bevorzugtes Nachrichtenformat 401 für die beschriebenen Ereignisse und/oder Nachrichten. Das Nachrichtenformat 401 umfaßt einen Zeitstempel 403, der Aufschluß über einen auszuführenden Zeitpunkt gibt und einen Adressaten 404, die Zielkomponente. Ferner ist ein Feld 402 vorgesehen, daß die Art der Nachricht klassifiziert. Handelt es sich um ein Ereignis, so ist dies in Feld 402 ebenso angezeigt, wie wenn es sich bei der Nachricht 401 um eine Benachrichtigung über eine Variablenänderung oder den Aufruf einer Funktion in einer Zielkomponente handelt.

Fig.5 zeigt eine Warteschlange 501 über einer Zeit t . Zu einem Zeitpunkt t_1 befinden sich drei Ereignisse 502 bis 504 in der Warteschlange, zu einem Zeitpunkt t_2 befindet sich nur noch das Ereignis 502 in der Warteschlange und zu einem Zeitpunkt t_3 befinden sich die Ereignisse 502 und 505 in der Warteschlange. Zu dem Zeitpunkt t_1 sind die Ereignisse 503 und 504 beendet worden, d.h. Ereignis 504 enthielt in dem Feld 403 (Zeitstempel) den Zeitpunkt t_1 , zu dem Zeitpunkt t_3 ist das Ereignis 505 neu in die Warteschlange eingetragen worden.

30

Fig.6 zeigt ein Drei-Tank-Beispiel für eine verfahrenstechnische Anlage zur Mischung von Flüssigkeiten. Die drei Tanks T1, T2 und T3 sind über Ventile V1, V2 und V3 miteinander verbunden, wobei in den Tanks T1 und T3 über Pumpen P1 und P2 Flüssigkeit S1 und S2 hinzugegeben werden kann. Über den Pfeil S3 wird der austretende Stoff-Fluß angezeigt.

35

Nachfolgend wird gezeigt, wie die Komponenten zur Modellierung der Anlage von Fig.6 aus Sicht der Prozeßmodellierung beschrieben werden. Mit Hilfe dieser Komponenten wird ein einfaches und funktionsfähiges Simulationsmodell für die in Fig.6 dargestellte Anlage erstellt. Dazu wird eine an die Programmiersprache C++ angelehnte Notation verwendet.

- 10 Einige wichtige Erläuterungen sind nachfolgend als Kommentare 701 bis 713, eingeleitet durch "//", in dem Quellcode vorgenommen. Die Beschreibungen zu den einzelnen Bezugszeichen 701 bis 713 erfolgt im Anschluß an den Quellcode.

15

QUELLCODE FÜR DREI-TANK-BEISPIEL:

```

ComponentLibrary n_tank {
20   Domain binary { range [0;1] }
      Domain auf_zu { set { zu, auf } }
      Domain notaus_typ { set { notaus, notaus_rueckname } }
25   Domain fehlermeldungen { set {pumpe_defekt,pumpe_ok } }
      Domain fuellgrad_typ { set { leer, halbvoll, voll } }
30   Domain schrittkette_tankanlage { set { bereit,
      vorbehandlungstank_fuellen, reinigungstank_fuellen, reinigen,
      reaktionstank_fuellen1 , reaktionstank_fuellen2, reagieren, leeren }
      }
35   Terminaltype volumenstrom { // 701
      process {
        inoutdata {
          druck: real default 1.0;
          durchfluss: real default 0.0;
40        }
      }
    }
45   Connectiontype volumenstrom_verbindung
      {
        terminals {
          ende_a: volumenstrom;
50          ende_b: volumenstrom;
        }
      }

```

16

```

behavior_descriptions {
  process {
    body{
      SelectionOfEquations(){$
5        SetBalance(ende_a.durchfluss,ende_b.durchfluss); // 702a
          SetIdentity(ende_a.druck,ende_b.druck);$ // 702b
      }
    }
  }
10 }

// 703
15 Interfacetype if_pumpe_sr2pm {
  inputevents {
    rueckmeldung: bool default false; // false: steht, true: laeuft
    q_ist : real default 0.0;
  }
20 outputsynchronousevents {
    q_soll : real default 0.0;
  }
}

25 Interfacetype if_tank_pm2sr {
  outputevents {
    fuellstand : real default 0.0;
  }
30 }

Interfacetype if_ventil_sr2pm {
  inputevents {
    rueckmeldung_auf: bool default false;
    rueckmeldung_zu: bool default true;
35 }
  outputsynchronousevents {
    zustand_soll : auf_zu default zu;
  }
40 }

...

45 Componenttype pumpe { // 704
  parameters {
    q_min: real unit kubikmeter_pro_sekunde;
    q_max: real unit kubikmeter_pro_sekunde;
50 }
  terminals {
    eingang: volumenstrom;
    ausgang: volumenstrom;
  }
55 behavior_descriptions { // 705
  control {
    ...
60 }

  process {

```

17

```

    interfaces {
        sr : if_pumpe_sr2pm inverted;
    }
    variables {
5      }
    body { // 706

        SelectionOfEquations() ($)
            SetBalance(eingang.durchfluss, ausgang.durchfluss);
10         SetExplicitEquation(eingang.durchfluss);
        $)
        G_Equations() ($)
            eingang.durchfluss = (double) sr.q_soll;
        $)
15     PostExecution() ($)
            if ( sr.q_soll > 0.0 )
                sr.rueckmeldung = true;
            else
                sr.rueckmeldung = false;
20         sr.q_ist = sr.q_soll;
        $)
    }
}
25 visualization { // 707
    . . .
}
    interface_connections {
30     . . .
    }
}

35

Componenttype ventil {
    parameters {
        widerstandsbeiwert: real default 1.0;
40     }
    terminals {
        eingang: volumenstrom;
        ausgang: volumenstrom;
    }
45     behavior_descriptions {
        control {
            . . .
        }
    }
    process {
50         interfaces {
            sr : if_ventil_sr2pm inverted;
        }
        variables {
            gl : real residue;
55         state_offen : bool disc_state;
        }
        body {

            InitSimulation() ($)
60             if( widerstandsbeiwert <= 0.0 ){
                Error("Value for parameter widerstandsbeiwert = %g not
                    valid", (double) widerstandsbeiwert);
            }
        }
    }
}

```

18

```

    )
    if ( !eingang.IsConnected() ) {
        Error("Terminal %z not connected",eingang.getFullName());
    }
5    if ( !ausgang.IsConnected() ) {
        Error("Terminal %z not connected",ausgang.getFullName());
    }
state_offen = false; // 708
    $)
10    InitDiscreteModel() ($)
        if ( (sr.zustand_soll.value() == auf_zu::auf)) {
            state_offen = true;
        }
15    if ( (sr.zustand_soll.value() == auf_zu::zu)) {
        state_offen = false;
    }
    $)
20    SelectionOfEquations() ($)
        SetBalance(eingang.durchfluss,ausgang.durchfluss);
        SelectGEquation(g1);
        if (state_offen) {
25    JacobiVariables(g1,&eingang.druck,&ausgang.druck,&eingang.durchfluss
        ,0);
        }
        else {
            JacobiVariables(g1,&eingang.durchfluss,0);
30    }
        $)

        G_Equations() ($)
            if (state_offen) {
35    g1 = eingang.durchfluss*widerstandsbeiwert - (eingang.druck-
            ausgang.druck); // 709
            }
            else {
                g1 = eingang.durchfluss;
40    }
            $)

        PostExecution() ($)
            if (state_offen) {
45    sr.rueckmeldung_auf = true;
                sr.rueckmeldung_zu = false;
            }
            else {
50    sr.rueckmeldung_auf = false;
                sr.rueckmeldung_zu = true;
            }
        $)
    )
55    visualization {
        . . .
    }
    )
    interface_connections {
60    control.pm = process.sr;
        . . .
    }

```

```

    }

Componenttype tank {
5   parameters {
        hoehe: real default 10.0 unit meter;
        anfangsfuellhoehe: real default 5.0 unit meter;
        grundflaeche      : real default 10.0 unit meter;
    }
10  terminals {
        eingang[0;1]: volumenstrom;
        ausgang[1;2]: volumenstrom;
    }
    behavior_descriptions {
15    control {
        . . .
    }
    process {
        interfaces {
20        sr : if_tank_pm2sr;
        }
        variables {
            fuellhoehe      : real cont_state default 0.0;
            d_fuellhoehe    : real diff_quot;
25        g[0;1]           : real residue;
            g_out[0;2]      : real residue;
        }
    body {
30        InitSimulation() ($ // 710
            g.set_size(eingang.size());
            g.notify(this,"g");
            g_out.set_size(ausgang.size());
            g_out.notify(this,"g_out");
            fuellhoehe.pm_init(0.0);
35        $)

        SelectionOfEquations() ($

            int i;
40            for (i=0; i<eingang.size(); i++) {
                if (eingang[i].IsConnected()) {
                    SelectGEquation(g[i]);
                    JacobiVariables(g[i],&eingang[i].druck,0);
                }
45            }

            for(i=0; i<ausgang.size(); i++){
                if (ausgang[i].IsConnected()) {
                    SelectGEquation(g_out[i]);
                    JacobiVariables(g_out[i],&ausgang[i].druck,0);
50                }
            }
            SelectFEquation(d_fuellhoehe,fuellhoehe,"fuellhoehe");
        $)
55        G_Equations() ($

            int i;
            double Bodendruck = fuellhoehe / 10.0 + 1.0;
60            for (i=0; i<eingang.size(); i++) {
                if (IsSelectedGEquation(g[i])) {

```

```

                                20
                                g[i] = eingang[i].druck - 1.0;
                                }
                                }

5      for (i=0; i<ausgang.size(); i++) {
        if (IsSelectedGEquation(g_out[i])) {
            g_out[i] = ausgang[i].druck - Bodendruck;
        }
    }
10     $)

    PostExecution() {$
        sr.fuellstand = fuellhoehe;
    $)

15     F_Equations() {$
        int i;
        double zufluss = 0.0;
        for (i=0; i<eingang.size(); i++) {
20         if (eingang[i].IsConnected()) {
            zufluss += eingang[i].durchfluss;
        }
    }

25     for (i=0; i<ausgang.size(); i++) {
        if (ausgang[i].IsConnected()) {
            zufluss += ausgang[i].durchfluss;
        }
    }
30     d_fuellhoehe = zufluss / grundflaeche; // 711
    $)

    }
    }
35     visualization {
        interfaces {
            sr      : if_tank_sr2vis inverted;
        }
    }
40     }
    interface_connections {
        control.pm = process.sr;
        control.vis = visualization.sr;
    }
45     }

Componenttype quelle_senke {
50     parameters {
        aussendruck: real default 1.0 unit bar;
    }
    terminals{
        einausgang: volumenstrom;
55     }
    behavior_descriptions {
        process {
            variables {
60                g1      : real residue; //residue;
            }

            body {

```



```

InitSimulation() {$
    if ( ! einausgang.IsConnected() ) {
        Error("Terminal %z not
5 connected",einausgang.getFullName());
    }
    $)

    SelectionOfEquations() {$
10     SelectGEquation(g1);
        JacobiVariables(g1,&einausgang.druck,0);
    $)

    G_Equations() {$
15     g1 = aussendruck - einausgang.druck;
    $)
}

20 }

Componenttype tank_anlage {
    parameters {
    }
    parts { // 712
        vorbehandlungstank : tank;
30     reinigungstank [1:] : tank;
        reaktortank : tank;
        ventil [1:number(reinigungstank)+1] : ventil;
        ausgangsventil : ventil;
35     pumpel: pumpe;
        pumpe2: pumpe;
        quelle1 : quelle_senke;
        quelle2 : quelle_senke;
        senke : quelle_senke;
    }
40     connections {
        quelle1.einausgang = pumpel.eingang;
        pumpel.ausgang = vorbehandlungstank.eingang[1];
        vorbehandlungstank.ausgang[1] = ventil[1].eingang;
        forall I in [1:number(reinigungstank)]
45     ventil[I].ausgang = reinigungstank[I].ausgang[1];
        forall I in [1:number(reinigungstank)]
            reinigungstank[I].ausgang[2] = ventil[I+1].eingang;
            ventil[number(reinigungstank)+1].ausgang =
50     reaktortank.ausgang[1];
            reaktortank.ausgang[2] = ausgangsventil.eingang;
            ausgangsventil.ausgang = senke.einausgang;
            quelle2.einausgang = pumpe2.eingang;
            pumpe2.ausgang = reaktortank.eingang[1];
        }
55     behavior_descriptions {

        control {
            . . . }

60     process {
    } // 713

```

22

```
visualization {  
    interfaces {  
        sr      : if_tankanlage_vis2sr ;  
    }  
5      }  
    }  
    interface_connections {  
        control.vis = visualization.sr;  
    }  
10    }  
    }
```

Die Bezugszeichen 701 bis 713 werden nachfolgend kurz
erläutert:

15

701: Definition der Schnittstelle für den Stoff-Fluß;

20

702a: "SetBalance" bewirkt, daß bei Verwendung dieser
Verbindung die beiden Schnittstellenwerte für den
Durchfluß dem Betrag nach identisch und sich im
Vorzeichen unterscheiden (die Bilanz ist dann gleich
Null);

25

702b: "SetIdentitiy" bewirkt eine Identifizierung der beiden
Druckwerte;

703: Definition der informationstechnischen Schnittstellen;

30

704: Definition der Komponententypen;

705: eine Funktionalität der Komponente, die außerhalb des
prozeßtechnischen Modells liegt;

35

706: Beschreibung des prozeßtechnischen Modells in
Segmenten;

707: Hinterlegung besonderer Visualisierungsinformation;

40

708: diskrete Modellierung des Ventilzustands
(offen/geschlossen);

709: g1 ist eine algebraische Gleichung, die dann gelöst ist, wenn der Wert von g1 Null wird;

5 710: Modellierungsmöglichkeiten, die sich mit Strukturinformationen und -auswertungen ergeben, sind in vorliegendem Beispiel nicht extra aufgeführt, das Segment "SettingProperties" wird daher nicht aufgenommen;

10 711: mit "d_fuellhoehe" wird der Differentialquotient für die Differentialgleichung zur Variable "fuellhoehe" beschrieben;

15 712: Festlegung der Komponenten und deren struktureller Beziehungen;

713: leeres Prozeßmodell.

Literaturverzeichnis:

- [1] U. Claussen: Objektorientiertes Programmieren - Mit Beispielen und Übungen in C++, Springer Verlag, Heidelberg 1993, ISBN 3-540-55748-2, Seiten 17-43.

Patentansprüche

1. Verfahren zur rechnergestützten Simulation eines technischen Systems, welches System mehrere Komponenten umfaßt,
 - a) bei dem jede der mehreren Komponenten mindestens eine Schnittstelle umfaßt,
 - b) bei dem jeder Komponente gemäß einer zu simulierenden Vorgabe eine Funktionalität einbeschrieben ist,
 - c) bei dem die Simulation durch ein Zusammenfügen der mehreren Komponenten durchgeführt wird, wobei über die mindestens eine Schnittstelle Information zwischen den Komponenten ausgetauscht wird und anhand eines Hauptprozesses ein Zusammenwirken der mehreren Komponenten gesteuert und ausgewertet wird.
2. Verfahren nach Anspruch 1, bei dem die Komponente ein Anlagenobjekt ist.
3. Verfahren nach Anspruch 1 oder 2, bei dem die Komponente eine Instanz einer Klasse eines objektorientierten Programms ist, durch welche Klasse der Typ des Anlagenobjekts beschrieben wird.
4. Verfahren nach einem der Ansprüche 1 bis 3,
 - a) bei dem die mindestens eine Schnittstelle eine erste Teilschnittstelle und eine zweite Teilschnittstelle umfaßt,
 - b) bei dem die erste Teilschnittstelle einen Eingang und einen Ausgang aufweist, wobei über den Eingang und den Ausgang der ersten Teilschnittstelle ein Stoff-Fluß modelliert wird,
 - c) bei dem die zweite Teilschnittstelle einen Eingang und einen Ausgang aufweist, durch welchen Eingang eine Veränderung einer Stellgröße der Komponente durchgeführt wird und durch welchen Ausgang eine Rückmeldung über einen Zustand des Stoff-Flusses

und/oder über einen Zustand der Veränderung der
Stellgröße ermittelt wird.

5. Verfahren nach einem der vorhergehenden Ansprüche,
5 bei dem die Komponente beschrieben wird durch mindestens
eine der folgenden Beschreibungsformen:
 - a) ein differentialalgebraisches System;
 - b) ein ereignisdiskretes Modell und
 - c) eine Strukturinformation.
- 10 6. Verfahren nach Anspruch 4 oder 5,
bei dem durch die erste Schnittstelle ein Prozeßverhalten
modelliert und durch die zweite Schnittstelle ein
Kontrollverhalten modelliert wird.
- 15 7. Verfahren nach einem der vorhergehenden Ansprüche,
bei dem die Komponente über mindestens einen der
folgenden Mechanismen Nachrichten erzeugt:
 - 20 a) von der Komponente wird ein Ereignis (Event) generiert
und an den Hauptprozeß übermittelt,
 - b) von der Komponente wird eine Variablenänderung für
eine andere Komponente generiert, indem die
Variablenänderung mit der Adresse der anderen
Komponente generiert und an den Hauptprozeß
25 übermittelt wird,
 - c) von der Komponente wird eine Methode einer anderen
Komponente aufgerufen, indem die Methode mit der
Adresse der anderen Komponente generiert und an den
Hauptprozeß übermittelt wird.
- 30 8. Verfahren nach Anspruch 7,
bei dem der Hauptprozeß folgende Schritte umfaßt:
 - a) das Ereignis wird in einer Warteschlange
abgespeichert,
 - 35 b) wenn das Ereignis bezogen auf eine
Simulationsrechenzeit fällig ist, wird dieses Ereignis
an den Adressaten weitergeleitet und dort ausgeführt.

9. Verfahren nach einem der Ansprüche 5 bis 8,
bei dem die Strukturinformation von dem Hauptprozeß
derart berücksichtigt wird, daß diese Strukturinformation
über Komponenten hinweg, entsprechend der Verbindung der
Komponenten, kommuniziert werden und somit
kontextabhängige Besonderheiten in der Simulation
aufgedeckt werden.
10. Verfahren nach einem der vorhergehenden Ansprüche,
bei dem die mindestens eine Schnittstelle jeder
Komponente derart ausgeführt ist, daß mehrere Komponenten
entsprechend ihrer zugrundeliegenden technischen
Bedeutung zusammengefügt werden.

1/3

FIG 1

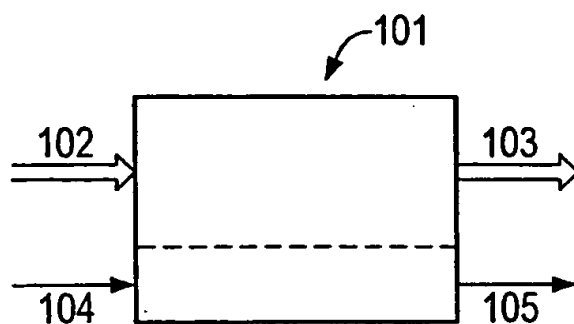
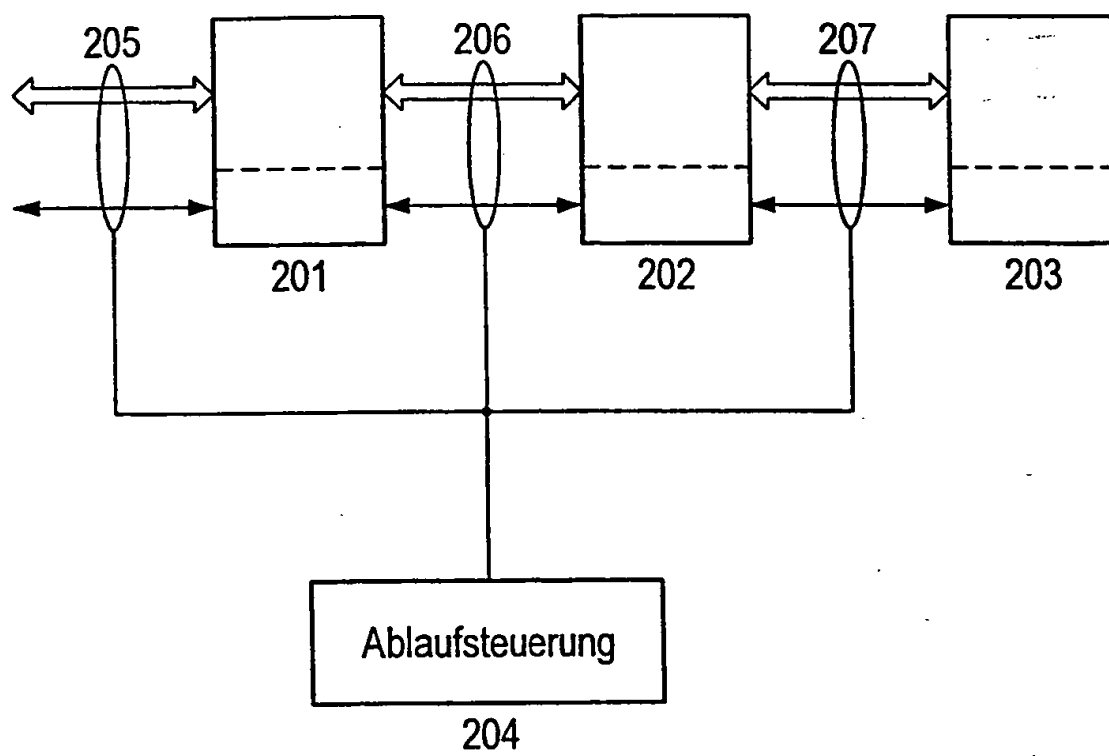
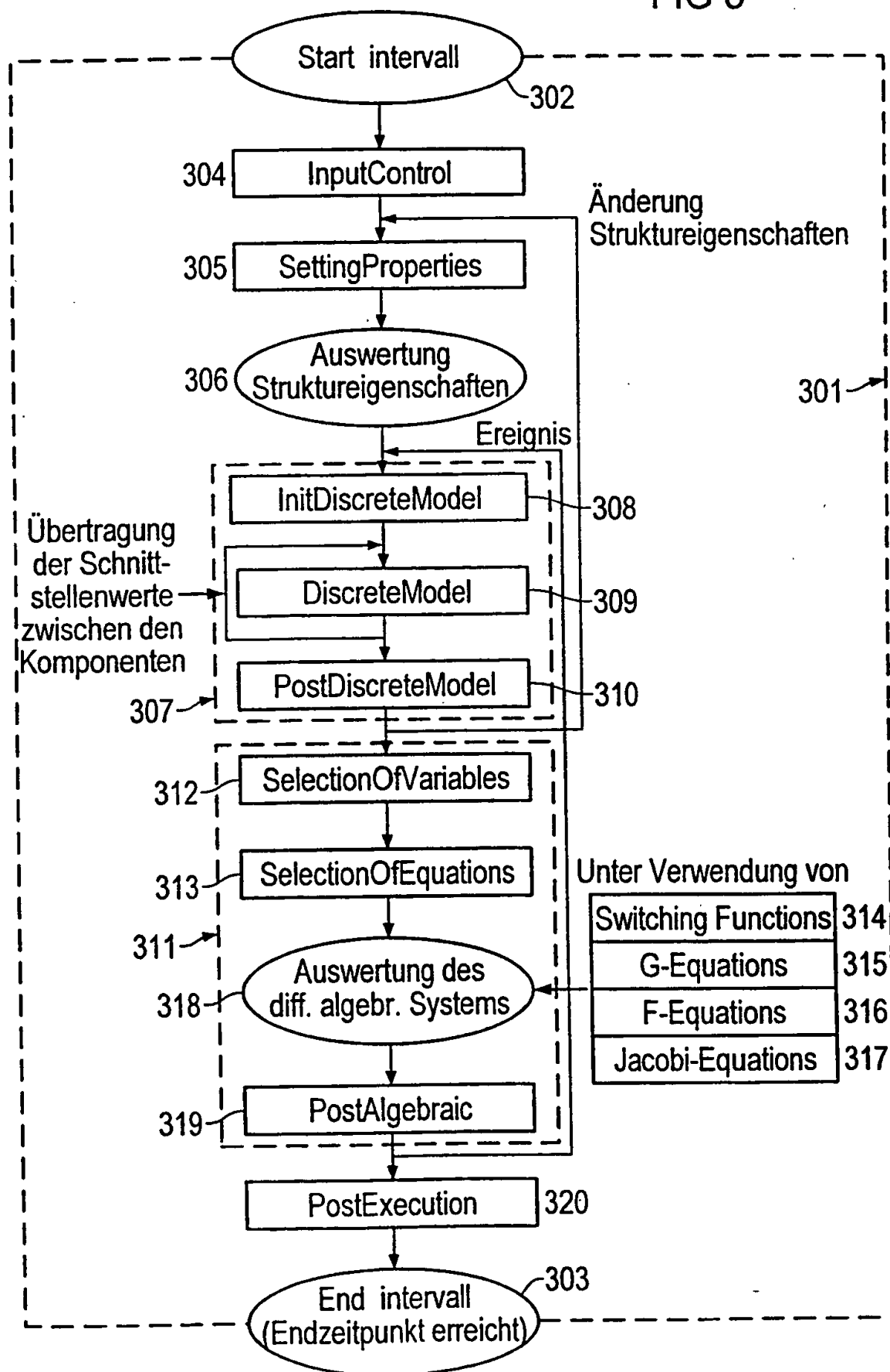


FIG 2



2/3

FIG 3



3/3

FIG 4



FIG 5

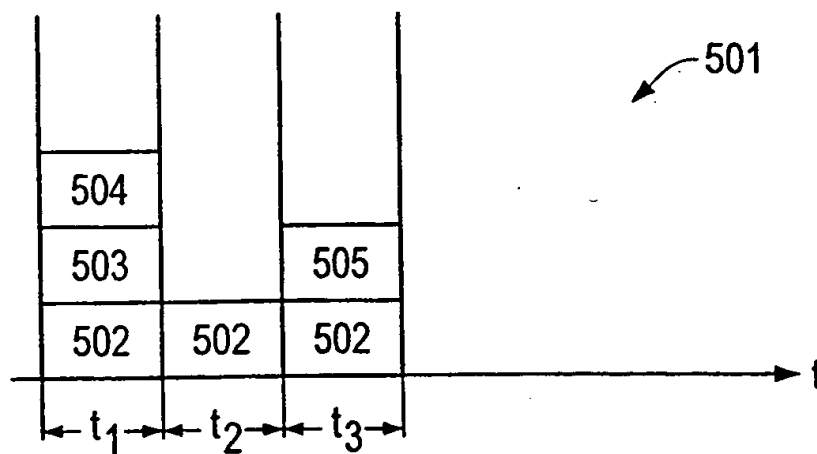
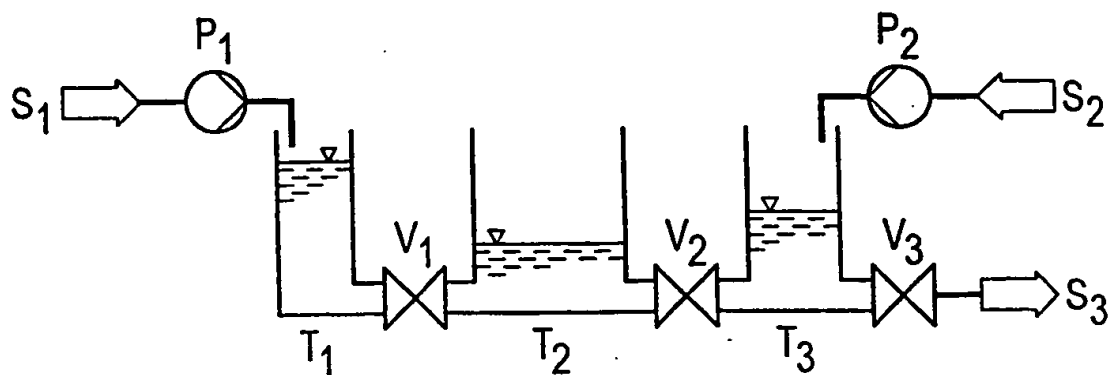


FIG 6



INTERNATIONAL SEARCH REPORT

International Application No

PCT/DE 99/01324

A. CLASSIFICATION OF SUBJECT MATTER

IPC 6 G06F17/50

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 572 733 A (RYU TADAMITSU ET AL) 5 November 1996 (1996-11-05) abstract; claims 1-6 column 1, line 25 -column 4, line 15; figures 1-4 column 5, line 55 -column 10, line 20; figures 5-9.	1-10
X	FR 2 724 744 A (ASS POUR LE DEV DE L ENSEIGNEM) 22 March 1996 (1996-03-22) page 4, line 9 -page 7, line 22 claims 1-12 page 9, line 20 -page 14, line 34; figures 2,3	1-10
X	US 5 331 579 A (MAGUIRE JR HAROLD T ET AL) 19 July 1994 (1994-07-19) abstract; claims 1-9	1



Further documents are listed in the continuation of box C.



Patent family members are listed in annex.

* Special categories of cited documents :

- "A" document defining the general state of the art which is not considered to be of particular relevance
- "E" earlier document but published on or after the international filing date
- "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed

- "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- "&" document member of the same patent family

Date of the actual completion of the international search

14 October 1999

Date of mailing of the international search report

21/10/1999

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl.
Fax: (+31-70) 340-3016

Authorized officer

Suendermann, R

INTERNATIONAL SEARCH REPORT

information on patent family members

International Application No

PC./DE 99/01324

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 5572733	A	05-11-1996	JP	6332712 A	02-12-1994
FR 2724744	A	22-03-1996	NONE		
US 5331579	A	19-07-1994	EP	0411873 A	06-02-1991
			JP	3071203 A	27-03-1991

A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
IPK 6 G06F17/50

Nach der Internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

B. RECHERCHIERTE GEBIETE

Recherchierte Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)
IPK 6 G06F

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
X	US 5 572 733 A (RYU TADAMITSU ET AL) 5. November 1996 (1996-11-05) Zusammenfassung; Ansprüche 1-6 Spalte 1, Zeile 25 - Spalte 4, Zeile 15; Abbildungen 1-4 Spalte 5, Zeile 55 - Spalte 10, Zeile 20; Abbildungen 5-9	1-10
X	FR 2 724 744 A (ASS POUR LE DEV DE L ENSEIGNEM) 22. März 1996 (1996-03-22) Seite 4, Zeile 9 - Seite 7, Zeile 22 Ansprüche 1-12 Seite 9, Zeile 20 - Seite 14, Zeile 34; Abbildungen 2,3	1-10
X	US 5 331 579 A (MAGUIRE JR HAROLD T ET AL) 19. Juli 1994 (1994-07-19) Zusammenfassung; Ansprüche 1-9	1



Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen



Siehe Anhang Patentfamilie

* Besondere Kategorien von angegebenen Veröffentlichungen :

"A" Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist

"E" älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist

"L" Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)

"O" Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht

"P" Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

"T" Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist

"X" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden

"Y" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist

"Δ" Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche

14. Oktober 1999

Absendedatum des internationalen Recherchenberichts

21/10/1999

Name und Postanschrift der internationalen Recherchenbehörde
Europäisches Patentamt, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Bevollmächtigter Bediensteter

Suendermann, R

INTERNATIONALER RECHERCHENBERICHT

Angaben zu Veröffentlichung: ja/nein (die zur selben Patentfamilie gehören)

Internationales Aktenzeichen

PCT/DE 99/01324

Im Recherchenbericht angeführtes Patentdokument		Datum der Veröffentlichung	Mitglied(er) der Patentfamilie		Datum der Veröffentlichung
US 5572733	A	05-11-1996	JP	6332712 A	02-12-1994
FR 2724744	A	22-03-1996	KEINE		
US 5331579	A	19-07-1994	EP	0411873 A	06-02-1991
			JP	3071203 A	27-03-1991